AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application. Claims 1-13, 14-16, 18 and 21 have been amended herein.

Listing of Claims:

1. (Currently amended): An application development system, comprising:

a development tool that facilitates application development in a design time environment and reports at least one of simulated run time and compile time information based upon design time attributes;

a software component; and

a type descriptor that accesses metadata associated with the software component, the type descriptor dynamically provides information associated with an instance of the software component to the development tool.

2. (Currently amended): The system of claim 1, ~~wherein~~ the type descriptor stores the information associated with the software component.

3. (Currently amended): The system of claim 1, ~~wherein~~ the information associated with the software component comprises at least one of types, members, attributes, properties and events.

4. (Currently amended): The system of claim 1, ~~wherein~~ the software component provides a custom type descriptor interface.

5. (Currently amended): The system of claim 4, ~~wherein~~ the custom type descriptor interface manipulates information associated with the software component and provides one of design time information and runtime information to the type descriptor.

6. (Currently amended): The system of claim 5, ~~wherein~~ the information provided to the development tool is one of the metadata and the manipulated information received from the custom type descriptor interface.

7. (Currently amended): The system claim 1, ~~wherein~~ the type descriptor provides the information associated with the software component in response to a request from a developer.

8. (Currently amended): An application development system, comprising:

a development tool that facilitates application development in a design time environment and reports at least one of simulated run time and compile time information based upon design time attributes; and,

a software component that provides a custom type descriptor interface which provides information associated with the software component to a type descriptor, the type descriptor accesses metadata associated with the software component ~~wherein~~ and the type descriptor receives manipulated information associated with the software component from the custom type descriptor interface and provides one of the metadata or information received from the custom type descriptor interface to the development tool.

9. (Currently amended): The system of claim 8, ~~wherein~~ the custom type descriptor interface manipulates the information associated with the software component to determine one of a design time and a runtime behavior of the software component before providing the information to the type descriptor.

10. (Currently amended): The system of claim 8, ~~wherein~~ the type descriptor stores one of the metadata or information received from the custom type descriptor interface.

11. (Currently amended): The system of claim 8, ~~wherein~~ the metadata comprises at least one of types, members, attributes, properties and events.

12. (Currently amended): The system of claim 8, ~~wherein~~ the information provided by the custom type descriptor interface comprises at least one of types, members, attributes, properties and events.

13. (Original): The system claim 8, where the type descriptor provides the information associated with the software component in response to a request from a developer.

14. (Currently amended): An application development system, comprising:

a plurality of contained components;

a development tool that reports at least one of simulated run time and compile time information based upon design time attributes to facilitate application development and that provides a container that facilitates communication among the plurality of contained components and further provides a site comprising a plurality of type descriptor filter service interfaces for manipulating information associated with the plurality of contained components; and

a type descriptor that ~~access~~ accesses metadata associated with the plurality of contained components ~~wherein~~ the type descriptor receives information associated with a design time behavior of the plurality of contained components from the plurality of type descriptor filter service interfaces and provides one of the metadata [[or]] and information received from the plurality of type descriptor filter service interfaces to the development tool ~~to facilitate application development~~.

15. (Currently amended): The system of claim 14 ~~wherein~~ at least one of the contained components provides a custom type descriptor interface.

16. (Currently amended): In a component based environment, a method for developing an application comprising:

determining whether an instance of a component implements a custom type interface;

invoking the custom type descriptor interface of the instance of the component, the custom type descriptor interface manipulating compile time information regarding the instance of the component ~~to determine~~ for simulating component behavior at one of a design time and a runtime ~~behavior of the component~~;

receiving the manipulated information regarding the instance of the component from the custom type descriptor interface; and,

storing the manipulated information regarding the instance of the component.

17. (Original): The method of claim 16, further comprising at least one of the following acts:

receiving a request from a development tool for information regarding the instance of the component;

discovering metadata associated with the instance of the component; and,

reporting the information regarding the instance of the component to the development tool.

18. (Currently amended): In a component based environment, a method for developing an application comprising:

receiving information regarding an instance of a component;

determining whether the instance of the component is contained by a container;

determining whether any other contained component desires to modify information regarding the instance of the component;

modifying the information regarding the instance of the component;

determining whether the container implements a type descriptor filter service interface for the instance of the component;

manipulating the compile time information regarding the instance of the component by the type descriptor filter service interface ~~to determine~~ for simulating component behavior at one of a design time and a runtime ~~behavior of the component~~; and,

storing the manipulated information regarding the instance of the component.

19. (Original): The method of claim 18, further comprising at least one of the following acts:

receiving a request from a development tool for information regarding the instance of the component;

discovering metadata associated with the instance of the component;

determining whether the instance of the component implements a custom type interface;

invoking the custom type descriptor interface of instance of the component, the custom type descriptor interface manipulating information regarding the instance of the component;

manipulating information regarding the instance of the component;

receiving information regarding the instance of the component from the custom type descriptor interface; and,

reporting the information regarding the instance of the component to the development tool.

20. (Original): A computer-readable medium having computer-executable instructions for executing at least a portion of the method of claim 16.

21. (Currently amended): An application development system comprising:

means for determining whether an instance of a component implements a custom type descriptor interface;

means for invoking the custom type descriptor interface of the instance of the ~~component wherein~~ component, the custom type descriptor interface ~~is adapted to manipulate~~ manipulates information regarding the instance of the component;

means for manipulating compile time information regarding the instance of the component ~~to determine~~ for simulating component behavior at one of a design time and a runtime ~~behavior of the component~~;

means for receiving the manipulated information regarding the instance of the component from the custom type descriptor interface; and,

means for storing the manipulated information regarding the instance of the component.

22. (Original): The application development system of claim 21, further comprising at least one of means for receiving a request from a development tool for information regarding the instance of the component, means for discovering metadata associated with the instance of the component and means for reporting the information regarding the instance of the component to the development tool.